

# Formatting Elements

<b>EAD3</b>	<b>EAC-CPF</b>	<b>EAD alignment decisions / suggestions</b>
<b>&lt;descriptivenote&gt;</b>	<b>&lt;descriptiveNote&gt;</b>	Adopt camel case in EAD4
<b>&lt;head&gt;</b>	<b>&lt;head&gt;</b>	EAC to add <head>, but without @althead or @altrender
@althead		<b>Recommendation:</b> Remove from EAD and to point (1) to @style for formatting according to CSS values and (2) to the option to include attributes from other namespaces
@altrender		<b>Recommendation:</b> Remove from EAD and to point (1) to @style for formatting according to CSS values and (2) to the option to include attributes from other namespaces? <b>Recommendation:</b> Retain in EAD (likely too disruptive to remove; often used by systems to output labels or qualifiers. See use in ASpace, esp. with <physdesc>)
<b>&lt;list&gt;</b>		
<head>	<head>	EAC to add <head>
<list>	<list>	EAD to add <list> as valid child element of <list>
<listhead>		EAC will not add <listhead>
@listType	@listType	Adopt camel case in EAD4; EAC will add @listType
deflist ordered unordered	--- ordered unordered	EAC will only permit values “ordered” and “unordered”, not “deflist” <b>Recommendation:</b> Remove “deflist” as a value, but create <defList> as a sibling element to <list>; this also means to remove <defItem> from <list> and only have it in the new <defList>
@mark	@style	<b>Recommendation:</b> Remove @mark and map to @style per EAC recommendation. Recommend use of W3C CSS attribute values (see also <a href="https://github.com/SAA-SDT/eac-cpf-schema/issues/171">https://github.com/SAA-SDT/eac-cpf-schema/issues/171</a> )
@numeration	@style	<b>Recommendation:</b> Remove @numeration and map to @style per EAC recommendation. Recommend use of W3C

		CSS attribute values (see also <a href="https://github.com/SAA-SDT/eac-cpf-schema/issues/171">https://github.com/SAA-SDT/eac-cpf-schema/issues/171</a> )
<defitem>		EAC will not add <defitem> <b>Recommendation:</b> Retain <defItem> in EAD4, but make it a sub-element of the new <defList>
<label>		<b>Recommendation:</b> Retain, but rename following the design principle of not using a name for both an attribute and element; <b>Suggestion:</b> use <term> which already has been inherited from EAC or rename <label> otherwise (e.g. HTML's "description term")
<item>		Retain
<item>	<item>	

## Mixed Content Elements

<u>EAD3</u>	<u>EAC-CPF</u>	<u>EAD alignment decisions / suggestions</u>
<abstract>	<abstract>	Available as child element of all descriptive elements (<did>)
Possible child elements of <abstract>: abbr, corpname, date, emph, expan, famname, footnote, foreign, function, genreform, geogname, lb, name, num, occupation, persname, quote, ptr, <b>reference</b> , subject, title	<reference> <span>	<p>EAD allows many mixed content children of &lt;abstract&gt; (but not &lt;p&gt;)</p> <p>EAC permits only &lt;reference&gt; and &lt;span&gt;</p> <p><b>Option 1:</b> In EAD, align with EAC-CPF and remove all mixed content children of &lt;abstract&gt;. &lt;abstract&gt; is intended to be a simplified / abridged version of &lt;scopecontent&gt; that can be shared across systems / serializations. Continue to allow mixed content in &lt;scopecontent&gt; but not &lt;abstract&gt;.</p> <p><b>Option 2:</b> Concede defeat and align with EAD: &lt;scopecontent&gt;. Would not allow direct mixed content children (e.g. &lt;persname&gt;), but would permit &lt;p&gt; children, which, in turn, allows mixed content children.</p> <p><b>Other notes:</b> EAD will rename &lt;ref&gt; to &lt;reference&gt; to align with EAC-CPF</p> <p><b>Recommendation:</b> Agreed to align with EAC and to reduce to &lt;reference&gt; and &lt;span&gt; (used with @localtype to capture e.g. "corpName" or "date"; used with @style to capture &lt;emph&gt; etc.)</p>

<event>	<event>	
<p>EAD allows the following mixed-content children of &lt;event&gt;: [text], abbr, corpname, date, emph, expan, famname, footnote, foreign, function, genreform, geogname, lb, list, name, num, occupation, persname, ptr, quote, <b>reference</b>, subject, title</p>	<p>&lt;reference&gt; &lt;span&gt;</p>	<p><b>Option 1:</b> In EAD4, align with EAC-CPF and remove all mixed content children of &lt;event&gt;, but add &lt;span&gt; and @style, which would provide formatting options.</p> <ul style="list-style-type: none"> <li>• Pros: simplicity / alignment, still permits formatting</li> <li>• Cons: disruptive</li> </ul> <p><b>Option2:</b> retain the current mixed content approach.</p> <ul style="list-style-type: none"> <li>• Pros: Easy migration path to EAD4 (do nothing)</li> <li>• Cons: maintains complexity; makes migration/systems integration difficult</li> </ul> <p><b>Recommendation:</b> Agreed to align with EAC and to reduce to &lt;reference&gt; and &lt;span&gt; (used with @localtype to capture e.g. “corpName” or “date”; used with @style to capture &lt;emph&gt; etc.)</p>
<item>	<item>	
<p>EAD allows the following mixed-content children of &lt;item&gt;: abbr, corpname, date, emph, expan, famname, footnote, foreign, function, genreform, geogname, lb, <del>list</del>, name, num, occupation, persname, ptr, quote, <b>reference</b>, subject, title</p>	<p>&lt;reference&gt; &lt;span&gt;</p>	<p>EAC will add &lt;reference&gt; as sub-element of &lt;item&gt; EAD will remove &lt;list&gt; as sub-element of &lt;item&gt; and rename &lt;ref&gt; to &lt;reference&gt;</p> <p><b>Option 1:</b> In EAD4, align with EAC-CPF and remove all mixed content children of &lt;item&gt;, but add &lt;span&gt; and @style, which would provide formatting options.</p> <ul style="list-style-type: none"> <li>• Pros: simplicity / alignment, still permits formatting</li> <li>• Cons: disruptive</li> </ul> <p><b>Option2:</b> retain the current mixed content approach.</p> <ul style="list-style-type: none"> <li>• Pros: Easy migration path to EAD4 (do nothing)</li> <li>• Cons: maintains complexity; makes migration/systems integration difficult</li> </ul> <p><b>Recommendation:</b> Agreed to align with EAC and to reduce to &lt;reference&gt; and &lt;span&gt; (used with @localtype to capture e.g. “corpName” or “date”; used with @style to capture &lt;emph&gt; etc.)</p>
<p>	<p>	
<p>EAD allows the following mixed-content children of &lt;p&gt;: [text], abbr, corpname, date, emph, expan, famname, footnote, foreign, function, genreform, geogname,</p>	<p>&lt;reference&gt; &lt;span&gt;</p>	<p><b>Option1:</b> For EAD4, retain the current mixed-content approach. There is probably heavy use of mixed-content children of &lt;p&gt; in the community. Removing mixed content elements from &lt;p&gt; would be too disruptive. For the most part, systems can ignore mixed-content children of &lt;p&gt; where necessary.</p> <p><b>Option2:</b> adopt EAC approach for &lt;span&gt; / @style for formatting, or allow @style in place of @render</p>

lb, list, name, num, occupation, persname, ptr, quote, <b>reference</b> , subject, title		<b>Recommendation:</b> Agreed to follow the same approach as outlined above, i.e. concentrating on only <span> and <reference>, and to get reaction from community on how disruptive (or not) such an approach would be
<emph>   <foreign>   <title>   <titleproper> @render	<span> @style	<p><b>Recommendation:</b> For EAD4, adopt EAC-CPF approach for formatting using &lt;span&gt; / @style in place of &lt;emph&gt; / @render</p> <p><i>Open question:</i> Consider allowing @style on other elements where @render is currently available: &lt;foreign&gt;, &lt;title&gt;, and &lt;titleproper&gt;?</p> <p><b>Recommendation:</b> follow the same approach as outlined above with regard to mixed content in general and to replace &lt;foreign&gt;, &lt;title&gt;, and &lt;titleproper&gt; with &lt;span&gt; using @localType</p>